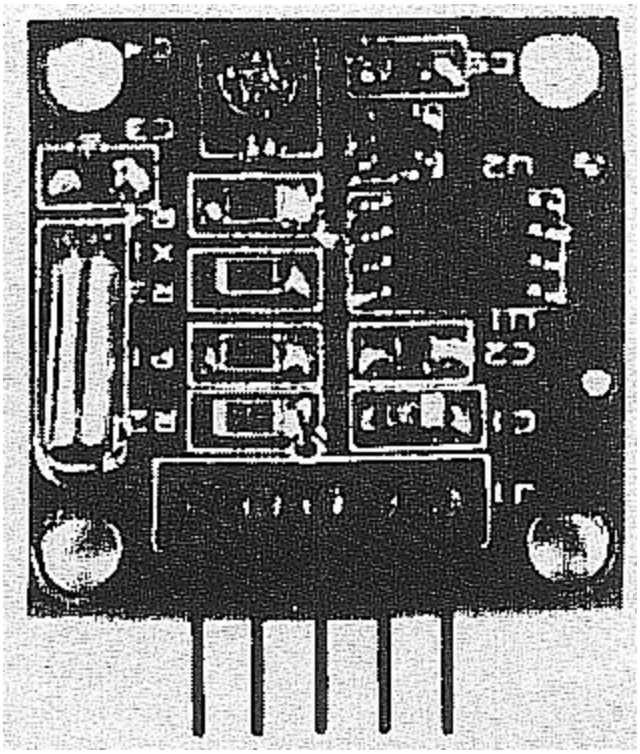
Jameco Part number 145630

P/N 145630



Pocket Watch B
Real Time Clock Module
*4 Alarm Types
*Easy Serial Interface

SOLUTIONS CUBED
3029-F Esplanade
Chico, CA 95973
phone (916) 891-8045
fax (916) 891-1643
www.solutions-cubed.com

MINIATURE ENGINEERING MODULES

Pocket Watch B

Serial Real Time Clock Module

FEATURES

- Real time clock: seconds, minutes, hours, days, months, years
- Leap year compensation
- **Year 2000 compliant**
- Easy to use 1 or 2 wire serial interface
- Four types of user configurable alarms
- Externally accessible, precision timebase
- Standard TTL levels
- No external components
- Easy to use SIP package

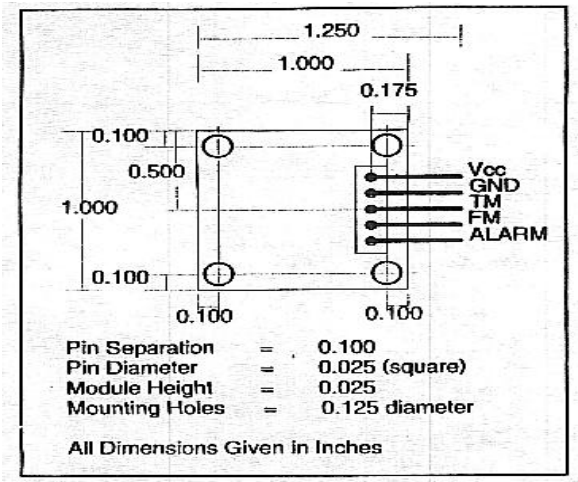
DESCRIPTION

The Pocket Watch B is an enhanced, pin for pin replacement of the Solutions Cubed Pocket Watch. It contains a real time clock, a calendar, and advanced timing features. The clock module keeps track of seconds, minutes, hours, days, months, and years. Adjustments for leap year are automatically performed. A year 2000 fix allows for worry-free use into the next millennium.

The Pocket Watch B communicates via an easy-to-use, asynchronous, two-wire, serial communications interface. Baud rates of 2400, 4800, and 9600 are supported with a user-friendly autobaud detect.

The Pocket Watch B contains four advanced timing features which are accessible with the alarm command. There is a standard level alarm; a single shot alarm with a duration of up to 18 hours; an astable alarm pulse with pulse lengths of up to 4 minutes and repetition rates of up to 4 hours; and an astable alarm pulse with pulse lengths of up to 4 hours and repetition rates of up to 10 days.

PIN CONFIGURATION AND MECHANICAL



SPECS

Vcc	Power supply pin
GND	Ground pin
TM	Serial Communication to master from Pocket Watch B. (open collector)
FM	Serial Communication from master to Pocket Watch B. (with weak pull up)
ALARM	Alarm output pin, active high, current limited

SPECIFICATIONS

ABSOLUTE MAXIMUM RATINGS

note: These are stress ratings only. Stresses above those listed below may cause permanent damage and/or affect device reliability. The operational ratings should be used to determine applicable ranges of operation.

Storage Temperature	-20°C to +70°C
Operating Temperature	-10°C to +60°C
Supply Voltage	0 to 7.0V
Voltage on any pin	-0.6V to (Vcc+0.6V)

DC ELECTRICAL CHARACTERISTICS

At T_A = 25°C and Vcc = 5.0V unless otherwise noted.

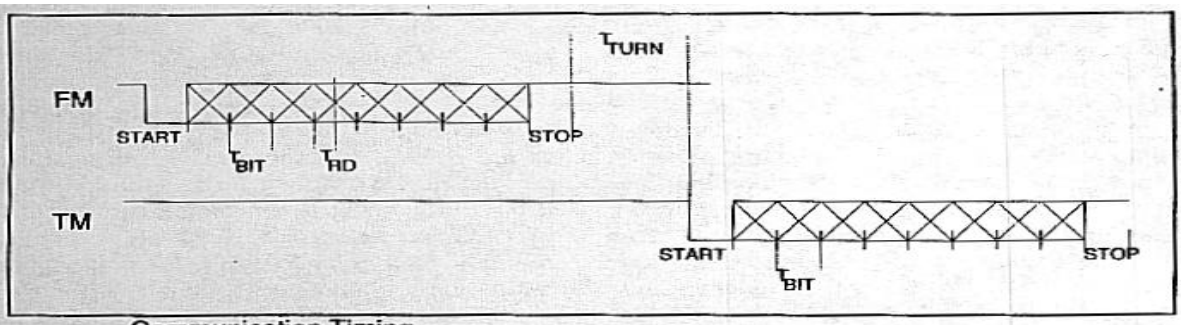
Characteristic	Symbol	Min	Typ	Max	Unit	Notes
Supply Voltage	Vcc	2.5		5.5	V	
Vcc rise time to ensure good reset	SVdd	0.05			V/ms	If this is not met, the Pocket Watch B may start up in an unknown state and may not communicate and/or keep time correctly.
Supply Current	Icc		3.5	6.0	mA	Alarm inactive, no communication
FM Input Low Voltage	VIL	GND		0.2Vcc	V	
FM Input High Voltage	VIH	2.0 0.2Vcc+1V		Vcc	V	4.0<Vcc<5.0 Full Vcc range User may use better of two specs.
FM Input Weak Pull Up current	IFMPU	50	250	400	µA	VFM = GND Min value is at VCC Min while Max value is at VCC max.
TM Output Low Voltage	VOLTM			0.6	V	
TM Output High Voltage	VIHTM	Vcc			V	TM is open collector
TM Output Pull Up current	ITMPU	2.5	5.0	5.5	mA	TM open collector is tied to Vcc with a 5% 1kΩ resistor.
ALARM pin Output Low Voltage	VOLA			0.6	V	ALARM pin has a 270Ω output impedance
ALARM pin Output High Voltage	VOHA	Vcc-0.7			V	ALARM pin has a 270Ω output impedance
ALARM pin Output current	IA			18.5	mA	ALARM shorted to ground
ALARM pin impedance	Ar		270		Ω	There is a series 5% 270 ohm resistor in line with the ALARM output.

note: "Typ" values are for design guidance only and are not guaranteed.

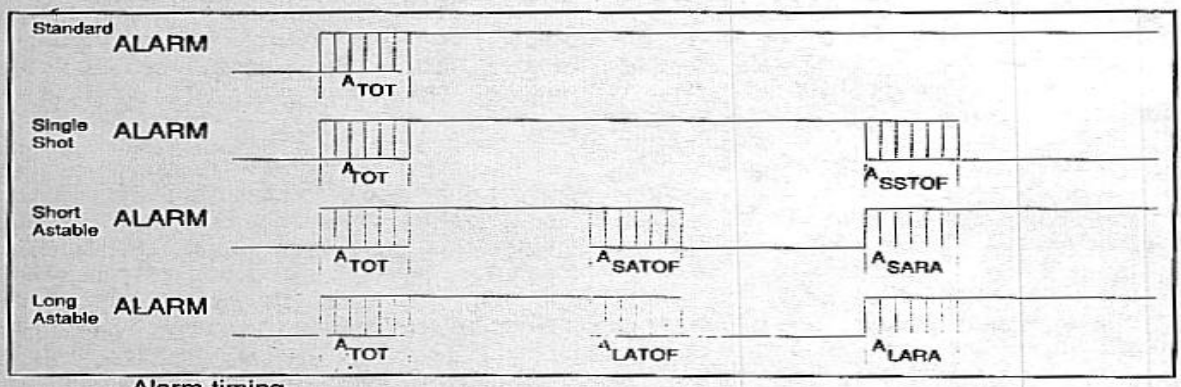
AC ELECTRICAL CHARACTERISTICS

At T_A = 25°C and Vcc = 5.0V unless otherwise noted.

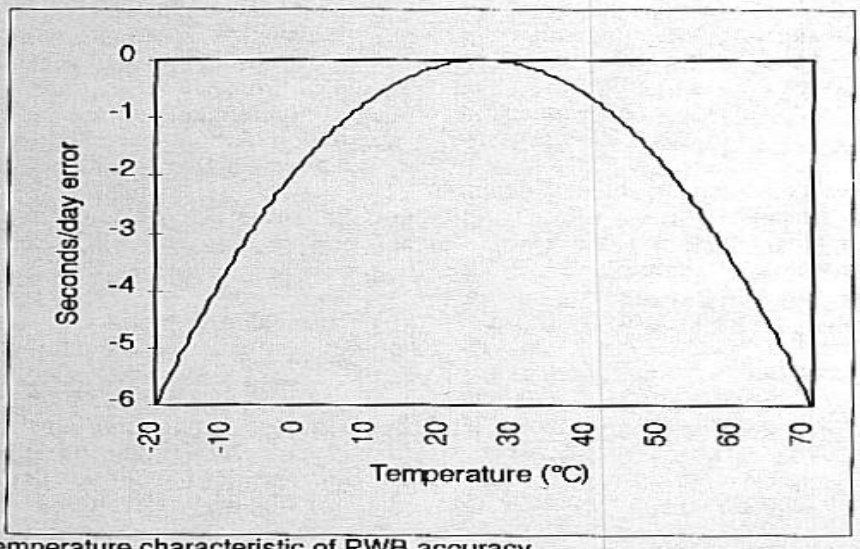
Characteristic	Symbol	Min	Typ	Max	Unit	Notes
Communication bit period 2400 4800 9600	TBIT	 413 206 103	 416 208 104	 419 211 105	µS	The bit period is determined by an on-board oscillator, and is temperature sensitive.
Offset when a bit is read 2400 4800 9600	TRD	180 90 45	200 100 50	220 110 60	µS	This is used to ensure a bit is valid when read. A bit must be valid for at least this long in order for the communication to not be erroneous.
Time for a command from master to be responded to	TTURN	450	500	550	µS	This time is used to allow for a master to change from transmission mode to reception mode.
Baud generator error	BGE		2	6	%	
Bit period temperature coefficient	BPTEMPCO	-1.8	-1.6	-1.7	NS/°C	Therefore at higher temperatures, a slower baud rate may be necessary at the master.
Time accuracy when clock is read	TSAC		±1		sec	The PWB can be up to 1 second slow if a communication happens just prior to a time update.
Time accuracy	TEAC		±154	±304	Sec/year	Because the timebase is factory calibrated, the major determining factor is the operating temperature of the PWB.
Alarm turn on time accuracy	ATOT			1	sec	Lowest resolution of clock
Single Shot turn off time accuracy	ASSTOF			1	sec	Lowest resolution of clock
Short Astable turn off time accuracy	ASATOF			1	sec	Lowest resolution of clock
Short Astable repetition accuracy	ASARA			1	min	Short astable alarm repeats on the nearest whole minute
Long Astable turn off time accuracy	ALATOF			1	min	Long astable alarm turns off at the nearest whole minute
Long Astable repetition accuracy	ALARA			1	min	Long astable alarm keeps track of hours to nearest minute



Communication Timing



Alarm timing



Temperature characteristic of PWB accuracy

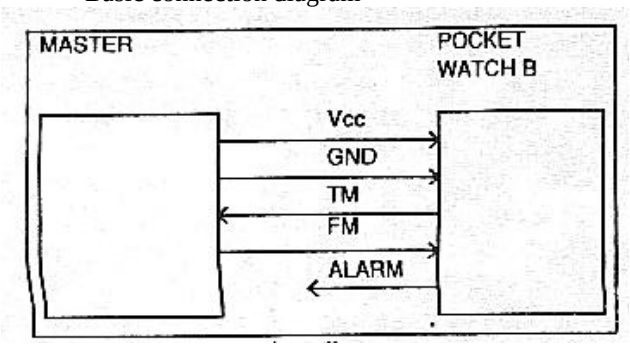
OPERATION

The Pocket Watch B is a time keeping module with advanced alarm features. Access to these features is based on a simple serial communications protocol. Because there are no external components necessary for use, the Pocket Watch is an extremely easy device to use.

Hardware Hook up

The connection diagram below shows the basic setup for using the Pocket Watch B. This is the easiest and simplest way to use the Pocket Watch B. Information about using the module with only one serial line is given in the **Applications** section of this data sheet.

Basic connection diagram



Power(Vcc) must be supplied to the Pocket Watch B from either a master processor or an external supply. When communication is taking place between the master and the Pocket Watch B, both the master's ground and the Pocket Watch B's GND pin must be at the same potential. The ALARM pin can either be read by the master or it can drive other circuitry. As the diagram shows, the TM pin on the Pocket Watch B provides the communication path to the master from the Pocket Watch B; while the FM pin on the Pocket Watch B provides the communication path from the master to the Pocket Watch B.

Time Format

The Pocket Watch B keeps track of the time in the following format: seconds, minutes, hours, days, months, years low, and years high. Each of these is represented as a complete and separate byte of data. Years low contains the last two digits of the year and years high contains the first two digits of the year. For example the year 1997 would be represented as '19' in years high and '97' in years low. With this system, the year 2000 century roll-over can be gracefully handled. The maximum date that this system can accommodate is 25599 before a roll over problem occurs.

All operations with the Pocket Watch operate with the seconds through years low data set, with the years high being an optional data byte (see the Command Structure section for more details).

The time of day is tallied in 24 hour time. For example, the hours value for 11pm would be '23', while 12am is '00'.

Leap years are automatically compensated, as are months having less than 31 days.

At power up, the Pocket Watch B initializes to 12:00am, January 1, 1997. The alarm set to 00:00am, 0/0/1997, which is a bogus time, so the alarm must be set prior to use.

Command Structure

note: all values are given in decimal unless otherwise noted.

Command	Byte Sent
Set time	'00'h
Set alarm	'01'h
Read time	'02'h
Read alarm	'03'h
Alarm on	'04'h
Alarm off	'05'h
Set time – extended	'10'h
Set alarm – extended	'11'h
Read time – extended	'12'h
Read alarm – extended	'13'h
Alarm on – extended	'14'h
Alarm off – extended	'15'h
Set single shot alarm	'16'h
Set short astable alarm	'17'h
Set long astable alarm	'18'h
Read alarm characteristics	'19'h

Table of Commands

The Pocket Watch B is a fully functional real time clock module which allows a master device to set time, set the alarm, read the time, read the alarm, turn an alarm on or off, set a particular alarm type, and read a particular alarm type.

The first four instructions allow access to the time registers through the years low. The alarm on and off commands allow access to the standard alarm. The `-extended' time commands (`10'h through `13'h) allow access to the year high register. The extended alarm functions (`14'h through `19'h) give access to the three special alarms in the Pocket Watch B. Any commands sent to the Pocket Watch which are not in the valid instruction range (`00'h through `05'h and `10'h through `19'h) will be ignored.

For differences between the command structure of the Pocket Watch B and the Pocket Watch see the **Differences** section of this data sheet.

The explanations below detail each of the commands shown in the table above. For information about implementing the commands, see the **Communication Protocol** section of this data sheet.

There are examples of using the command set at the end of this section.

Set time (`00'h) When setting the time the following values must be sent to the FM pin of the Pocket Watch B: seconds, minutes, hours, days, months, and years low. All of the values must be sent or the time will not be set correctly. If an out of range value is sent, that byte is reset internally in the Pocket Watch B. For example, if the days byte was sent as `53' the Pocket Watch B would place a `1' in its internal days byte.

Set alarm (`01'h) This command sets the time at which an alarm will go off if enabled. After setting an alarm, the alarm must be activated by either of the `Turn on alarm' commands. When setting the alarm the following values must be sent to the FM pin of the Pocket Watch B: seconds, minutes, hours, days, months, and years low. All of the values must be sent or the alarm will not be set correctly. If an out of range value is sent, that byte is reset internally in the Pocket Watch B. For example, if the alarm days byte was sent as `53' the Pocket Watch B would place a `1' in its internal alarm days byte.

Read time (`02'h) When reading the time, the following bytes will be sent to the master system from the Pocket Watch B via the TM pin in the following order: seconds, minutes, hours, days, months, and years low.

Read alarm (`03'h) When reading the time at which the alarm is set to activate, the following bytes will be sent to the master system from the Pocket Watch B via the TM pin in the following order: seconds, minutes, hours, days, months, and years low.

Turn on alarm (`04'h) This command will enable the alarm. The alarm must be enabled by this command to be active. The ALARM pin will toggle from a low to a high when the time in the Pocket Watch B equals the time which was set by the `set alarm' command.

Turn off alarm (`05'h) This command has two purposes. The first is to turn off any alarm which has not been activated. In this case, when the time in the Pocket Watch B equals the time set by the `set the alarm' command, no action will be taken by the Pocket Watch B. The second purpose is to turn off any alarm which has been activated. At this point the ALARM output will toggle from a high to a low. All extended alarms are also de-activated.

Set time - extended (`10'h) When setting the extended time the following values must be sent to the FM pin of the Pocket Watch B: seconds, minutes, hours, days, months, years low, and years high. All of the values must be sent or the time will not be set correctly. If an out of range value is sent, that byte is reset internally in the Pocket Watch B. For example, if the days byte was sent as `53' the Pocket Watch B would place a `1' in its internal days byte.

Set alarm - extended (`11'h) This command sets the time at which an alarm will go off if enabled. After setting an alarm, the alarm must be activated by either of the `Turn on alarm' commands. When setting the alarm with this command the following values must be sent to the FM pin of the Pocket Watch B: seconds, minutes, hours, days, months, years low, and years high. All of the values must be sent or the alarm will not be set correctly. If an out of range value is sent, that byte is reset internally in the Pocket Watch B. For example, if the alarm days byte the TM pin in the following order: seconds, minutes,

hours, days, months, years low, and years high.

Read time - extended (`12'h) When reading the time with this command, the following bytes will be set to the master system from the Pocket Watch B via the TM pin in the following order: seconds, minutes, hours, days, months, years low, and years high.

Read alarm - extended (`13'h) When reading the time at which the alarm is set to activate with this command, the following bytes will be sent to the master system from the Pocket Watch B via the TM pin in the following order: seconds, minutes, hours, days, months, years low, and years high.

Turn on alarm - extended (`14'h) This command will enable the alarm. If a special alarm has been set, then this command will enable that alarm as opposed to the standard alarm.

Turn off alarm - extended (`15'h) This command operates exactly as does the other 'turn off alarm `05'h does.

Set single short alarm (`16'h) This command sets the duration of the single shot alarm. The single shot alarm can be set for a duration of 1 through 65536 seconds in increments of 1 second. After the `16'h command is sent, two further bytes are immediately sent which set the duration of the single shot alarm. For example, to turn the alarm on for 1000 seconds the following bytes would be sent: (`55'h `16'h `03'h `E7'h). Further discussion of the single shot alarm can be found in the **Alarms** section of this data sheet

Set short astable alarm (`17'h) This command sets the duration of the alarm pulse and its repetition rate in an astable mode. The pulse length can be from 1 to 256 seconds long and the repetition rate can be from 1 to 256 minutes long. Immediately after the `set short astable alarm' command is sent, the repetition rate, in minutes, is sent (1 byte) followed by the duration, in seconds, (1 byte). For example to have the ALARM pin turn on every 15 minutes for 30 seconds the following bytes would be sent: (`55'h `17'h `0E'h `1D'h).

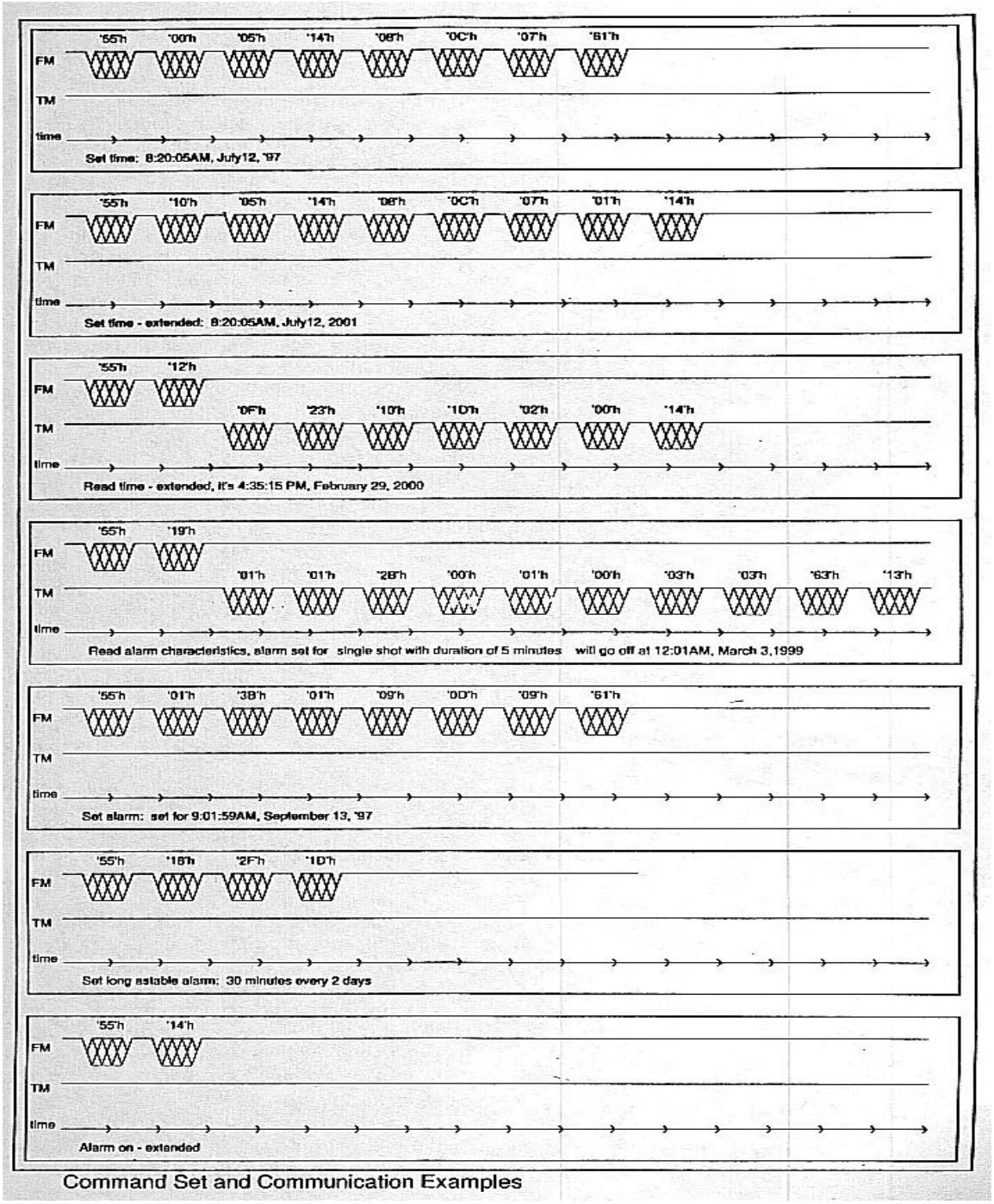
Further discussion of the short astable alarm can be found in the **Alarms** section of this data sheet

Set long astable alarm (`18'h) This command sets the duration of the alarm pulse and its repetition rate in an astable mode. The pulse length can be from 1 to 256 minutes long and the repetition rate can be from 1 to 256 hours long. Immediately after the `set long astable alarm' command is sent, the repetition rate, in hours, is sent (1 byte) followed by the duration, in minutes, (1 byte). For example to have the ALARM pin turn on every 24 hours for 15 minutes the following bytes would be sent: (`55'h `18'h `17'h `0E'h). Further discussion of the long astable alarm can be found in the **Alarms** section of this data sheet

Read Alarm Characteristics (`19'h) When the Pocket Watch B receives this command it sends three bytes of information containing all of the information about the alarm settings, followed by the time (extended) that the alarm is set to turn on. The first byte sent back tells which alarm is set. The second byte gives one of four values depending on which alarm is set. The third byte sent back gives one of four values depending on which alarm is set. The table below shows what the bytes mean. After the first three bytes, the information sent back conforms directly to the `read alarm - extended (`13h') command.

Byte 1	**Byte 2**	**Byte 3**
`00'h regular alarm	Don't care	don't care
`01'h single shot alarm	High byte of duration	low byte of duration
`02'h short astable alarm	Repetition rate (minutes)	duration (seconds)
`03'h long astable alarm	Repetition rate (hours)	duration (minutes)

Meanings of first three bytes returned with a `read alarm characteristics' command


'55'h '00'h '05'h '14'h '08'h '0C'h '07'h '61'h
FM
TM
time
Set time: 8:20:05AM, July12, '97
'55'h '10'h '05'h '14'h '08'h '0C'h '07'h '01'h '14'h
FM
TM
time
Set time - extended: 8:20:05AM, July12, 2001
'55'h '12'h
FM
'0F'h '23'h '10'h '1D'h '02'h '00'h '14'h
TM
time
Read time - extended, it's 4:35:15 PM, February 29, 2000
'55'h '19'h
FM
'01'h '01'h '28'h '00'h '01'h '00'h '03'h '03'h '63'h '13'h
TM
time
Read alarm characteristics, alarm set for single shot with duration of 5 minutes will go off at 12:01AM, March 3,1999
'55'h '01'h '3B'h '01'h '09'h '0D'h '09'h '61'h
FM
TM
time
Set alarm: set for 9:01:59AM, September 13, '97
'55'h '18'h '2F'h '1D'h
FM
TM
time
Set long astable alarm: 30 minutes every 2 days
'55'h '14'h
FM
TM
time
Alarm on - extended
Command Set and Communication Examples

COMMUNICATION PROTOCOL

Communication with the Pocket Watch B is accomplished with a two-wire (labeled TM and FM), asynchronous, serial communication channel. The FM pin carries data and commands from the master device to the Pocket Watch B. The TM pin carries data and commands to the master device from the Pocket Watch B.

All communication is 8N1, least significant bit first, 1 start bit, and 1 stop bit. The Pocket Watch B can accommodate 2400, 4800, and 9600 baud. All communication must be initiated by the master processor. The Pocket Watch B cannot initiate communication.

Every communication must be started with a '55'h sync byte. This allows the Pocket Watch B to automatically detect the baud rate. After this sync byte, the command may be sent along with any additional information, if necessary. The Pocket Watch B will ignore all incoming data, until it sees a '55'h sync byte. Responses from the Pocket Watch B to the master do not use the sync byte.

Any response that the Pocket Watch B sends to the master will be at the same baud rate at which the most recent command was received. For example if a 'read the time - extended' command was received at 9600 baud, then the Pocket Watch B would send all of the time information from seconds through years high at 9600 baud.

Some examples showing the use of the communication protocol and the command set are given on page 8 of this data sheet.

ALARMS

The Pocket Watch B contains four different alarm types: standard, single shot, short astable, and long astable. With these four alarm types, the Pocket Watch B can be configured for numerous alarm and timer functions.

Each of the alarms is exclusive; therefore, if one alarm is selected the other three cannot be used. When a new alarm type is selected, the settings from the old alarm setting are lost.

Standard Alarm This is a standard alarm level toggle, much like a common alarm clock. When the values in the alarm registers equal the time, as kept by the Pocket Watch B, the ALARM pin will toggle from low to high. The ALARM pin will remain high indefinitely. The only way to have the ALARM pin go back low is to remove power from the Pocket Watch B or use either of the `turn alarm off' commands.

Single Shot Alarm This command sets a duration for the alarm to remain active then turn off. This allows the Pocket Watch B to toggle the ALARM pin on for a set period and then turn off the ALARM pin with no action taken by the master processor. The Pocket Watch B will only perform this sequence once, after the alarm has turned off, the master must reset the alarm if another single shot is wanted.

The duration of the single shot has a resolution of seconds and is set by two bytes: DURATION_HIGH and DURATION_LOW which are concatenated to make one 16 bit value. In this manner durations of 1 to - 65536 seconds can be achieved. The duration lasts for one second longer than the loaded value. For example if a duration of 2 minutes (120 seconds) was wanted, DURATION_HIGH would be loaded with `00'h and DURATION_LOW would be loaded with `77'h.

Short Astable Alarm When this alarm is used the ALARM pin will toggle high at the time that the alarm is set to go off. From this point on, the ALARM pin will act like a timer, toggling on for an amount of time and then off, repeating this sequence indefinitely. The amount of time the ALARM will be high is set by one byte: DURATION; the repetition rate is set by one byte: PERIOD. DURATION can have a value of 1 to 256 seconds while PERIOD can have a value of 1 to 256 minutes. Each of these values is incremented by one, so the shortest duration is 1 second while the shortest repetition rate is 1 minute. For example if the ALARM pin should be on for 3 minutes and repeat every 15 minutes DURATION would be loaded with ‘B3’h and PERIOD would be loaded with ‘0E’h.

Long Astable Alarm When this alarm is used the ALARM pin will toggle high at the time that the alarm is set to go off. From this point on, the ALARM pin will act like a timer, toggling on for an amount of time and then off, repeating this sequence indefinitely. The amount of time the ALARM will be high is set by one byte: DURATION; the repetition reate is set by one byte: PERIOD. DURATION can have a value of 1 to 256 minutes while PERIOD can have a value of 1 to 256 hours. Each of these values is incremented by one, so the shortest duration is 1 minute while the shortest repetition rate is 1 hour. For example if the ALARM pin should be on for 90 minutes and repeat every 24 hours DURATION would be loaded with '59'h and PERIOD would be loaded with '17'h.

OSCILLATOR CALIBRATION

Solutions Cubed can make no guarantee to the accuracy or even the operation of the Pocket Watch B once an end user calibrates the Pocket Watch B, Calibration should be attempted only by those using the correct tools.

The timebase is based on a standard 32.768kHz crystal circuit which has been factory trimmed. This allows the Pocket Watch B to achieve a very steady timebase over time. For most normal use, user calibration should never be necessary. However, due to component aging, temperature stresses, and/or operation in harsh environments, the oscillator frequency may drift. If this happens, the oscillator may need to be trimmed by the end user.

Accessing the oscillator output at the "TP2" point located on the Pocket Watch B circuit board, the trim capacitor C4 may be adjusted until the frequency out of the oscillator circuit is exactly 32.768kHz. Once the output is 32.768kHz exactly for the required operating environment, the Pocket Watch B should meet all of the timing specifications given in this data sheet.

DIFFERENCES BETWEEN POCKET WATCH AND POCKET WATCH B

The Pocket Watch B is a direct pin for pin replacement for the original Pocket Watch module provided by Solutions Cubed. The Pocket Watch B contains many advanced features which the original Pocket Watch does not. The list below details the upgrades which the Pocket Watch B has.

Pocket Watch B Upgrades

- Factory calibrated timebase
- User accessible timebase
- User calibration of timebase
- 3 advanced alarms: single shot, short astable, long astable
- Year 2000 fix
- PCB insertable SIP connector
- Streamlined communication protocol

Converting Master Software

The basic command set from the original Pocket Watch has been implemented directly into the Pocket Watch B, therefore any program which works with the Pocket Watch will work with the Pocket Watch B with only one modification.

The 2mS attention pulse which the original Pocket Watch required to start communication is no longer necessary. Under most circumstances the Pocket Watch B will ignore the attention pulse, however communication errors may arise if left in. In order to guarantee communication, the attention pulse should be removed from any old Pocket Watch program accessing the Pocket Watch B.

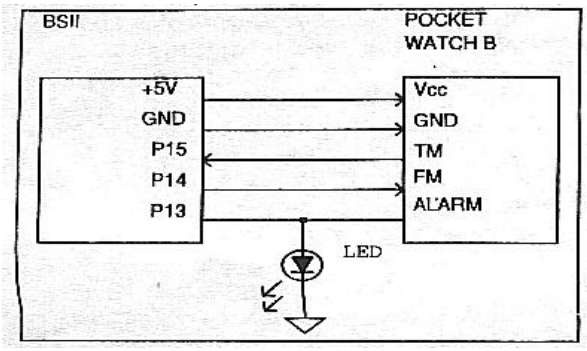
APPLICATIONS

The following examples show how to interface the Pocket Watch B to various master processors in various configurations. These examples should make it fairly easy for an end user to custom design their own programs and uses for the Pocket Watch B.

AN-151 'Interfacing to a Parallax BASIC Stamp II', AN-152 'Interfacing with a single wire', and the hardware diagram for AN-153 'Interfacing to a PC' are all included here.

AN-151 Interfacing to a Parallax BASIC Stamp II

The Parallax BASIC Stamp II makes an ideal choice for the master processor for the Pocket Watch B for one main reason: ease of developing code. The SERIN and SEROUT commands which the BASIC Stamp interpreter supports, provide the end user with an effective and simple method to interface to the Pocket Watch B. The diagram below shows the connection diagram used for this application.



The Pocket Watch B's ALARM pin is connected is the Basic Stamp II so that the BSII can tell when an alarm has happened. The LED gives a visual indicator of when an alarm has gone off. Because the ALARM pin is internally current limited, there is no need to put a current limit resistor in line with the LED.

The program below simply goes through all of the commands and alarms. The long astable command is not included here; however, using it is exactly like using the short astable alarm. The in-line comments, should make it easy to understand.

AN-151 Interfacing to a BSII code listing

```
'AN-151 BSII Interface to Pocket Watch B 'by Solutions Cubed '07/97
'Set I/O pin directions
input 15                                                    'communication FROM Pocket Watch B
output 14                                                   'communication TO Pocket Watch B
input 13                                                    'ALARM monitoring pin
'Declare variables
ss                  var     byte                            'seconds
mm                  var     byte                            'minutes
hh                  var     byte                            'hours
dd                  var     byte                            'days
mo                  var     byte                            'months
yl                  var     byte                            'years low
yh                  var     byte                            'years high
Begin:
HIGH                14                                      'ensure no spurious start bit
PAUSE               1000
SetTimeCommand:                                             'set to 6:30:00AM, June 3, 1997
                    SEROUT  14,84,[$55,$00,$00,$1E,$06,$03,$06,$61]
SetAlarmCommand:                                            'set to 15 seconds after time
                    SEROUT  14,84,[$55,$01,$0F,$1E,$06,$03,$06,$61]
ReadAlarmCommand:                                           'see what time alarm set to go off
                    SEROUT  14,84,[$55,$03]
                    SERIN   15,84,5000,BadAlarm1,[ss,mm,hh,dd,mo,yl]
                    DEBUG   "Alarm: ",dec2 hh,":",dec2 mm,":",dec2 ss," ",dec2 mo,"/",dec2 dd,"/19",dec2 yl,cr
```

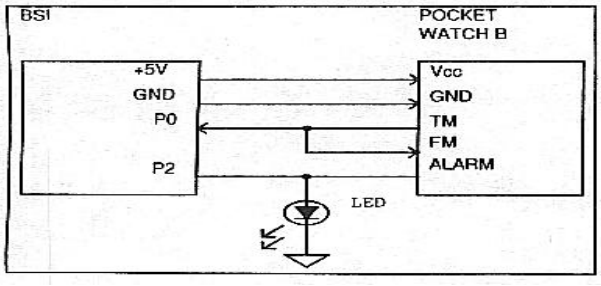
```
AlarmOnCommand:                                                                 'enable alarm
        SEROUT          14,84,[$55,$04]
Loop1:
        PAUSE           2000                                                    'wait 2 seconds between reads
ReadTimeCommand:                                                                'see what time it is presently
        SEROUT          14,188,[$55,$02]
        SERIN           15,188,5000,BadLoop1,[ss,mm,hh,dd,mo,yl]
        DEBUG           "Time: ",dec2 hh,":",dec2 mm,":",dec2 ss," ",dec2 mo,"/",dec2 dd,"/19",dec2 yl, cr
CheckALARM:                                                                     'see if ALARM is high
        IF IN13=1 THEN AlarmOff1                                                If ALARM not high read time
        GOTO            LOOP1
AlarmOff1:
        DEBUG           "Regular Alarm On",CR
        PAUSE           5000                                                    'allow alarm to stay of for 5 sec
AlarmOffCommand:                                                                'turn off alarm
        SEROUT          14,188,[$55,$05]
        DEBUG           "Regular Alarm Off",CR,CR
SetTimeExtended:                                                                'set time to 9:45:00PM, November 16, 2001
        SEROUT          14,396,[$55,$10,$00,$2D,$15,$0B,$10,$01,$14]
SetAlarmExtended:                                                               'set alarm for 15 seconds later
        SEROUT          14,396,[$55,$11,$0F,$2D,$15,$0B,$10,$01,$14]
ReadAlarmExtended:                                                              'see what time alarm is on
        SEROUT          14,396,[$55,$13]
        SERIN           15,396,5000,BadAlarm2,[ss,mm,hh,dd,mo,yl,yh]
        DEBUG           "Alarm: ",dec2 hh,":",dec2 mm,":",dec2 ss," ",dec2 mo,"/",dec2 dd,"/",dec2 yh,dec2 yl,cr
SetSingleShot:                                                                  'set singleshot  10 seconds
        SEROUT          14,84,[$55,$16,$00,$09]
AlarmOnExtended1:                                                               'enable alarm
        SEROUT          14,84,[$55,$14]
ReadAlChar1:                                                                    'get alarm characteristics
        SEROUT          14,84,[$55,$19]
        SERIN           15,84,5000,BadAlarm3,[b10,b11,b12,ss,mm,hh,dd,mo,yl,yh]
        DEBUG           "Alarm type: ",dec2 b10,cr,"Byte 1: ",dec2 b11,cr,"Byte 2: ",dec2 b12,cr
        DEBUG           "Alarm: ",dec2 hh,":",dec2 mm,":",dec2 ss," ",dec2 mo,"/",dec2 dd,"/",dec2 yh,dec2 yl, cr
Loop2:
        PAUSE           2000                                                    'walt 2 seconds between reads
ReadTimeEx1:                                                                    'read extendled time
        SEROUT          14,396,[$55,$12]
        SERIN           15,396,5000,BadReadex1,[ss,mm,hh,dd,mo,yl,yh]
        DEBUG           "Time: ",dec2 hh,":",dec2 mm,":",dec2 ss," ",dec2 mo,"/",dec2 dd,"/",dec2 yh,dec2 yl, cr
CheckSingleShot1:                                                               'see if single shot started
        IF IN 13=1 THEN CheckSingleShot2                                        'if single shot not started read time
        GOTO            LOOP2
CheckSingleShot2:
DEBUG                   "Single Shot Started",CR
Loop3:
        IF IN 13=0 THEN CheckSingleShot3                                        'see if single shot over
        GOTO            Loop3
CheckSingleShot3:
        DEBUG           "Single Shot Done",CR,CR
SetShortAsable:                                                                 '15 seconds every 1 minute
        SEROUT          14,84,[$55,$17,$00,$0E]
SetAlarmEx2:
        SEROUT          14,84,[$55,$11,$00,$2E,$15,$0B,$10,$01,$14]
AlarmOnEx2:                                                                     'turn on short astable
        SEROUT          14,188,[$55,$14]
ReadAlChar2:                                                                    'get alarm characteristics
        SEROUT          14,84,[$55,$19]
        SERIN           15,84,5000,BadAlarm3,[b10,b11,b12,ss,mm,hh,dd,mo,yl,yh]
        DEBUG           "Alarm type: ",dec2 b10,cr,"Byte 1: ",dec2 b11,cr,"Byte 2: ",dec2 b12,cr
        DEBUG           "Alarm: ",dec2 hh,":",dec2 mm,":",dec2 ss," ",dec2 mo,"/",dec2 dd,"/",dec2 yh,dec2 yl, cr
ReadTimeEx2:                                                                    'read extendted time
        PAUSE           10000                                                   'wait 10 seconds between reads
        SEROUT          14,396,[$55,$12]
        SERIN           15,396,5000,BadReadex2,[ss,mm,hh,dd,mo,yl,yh]
        DEBUG           "Time: ",dec2 hh,":",dec2 mm,":",dec2 ss," ",dec2 mo,"/",dec2 dd,"/",dec2 yh,dec2 yl, cr
CheckAstable1:                                                                  'see if astable started
        IF IN13=1 THEN CheckAstable2                                            'if single shot not started read time
        GOTO            ReadTimeEx2
```

```
                DEBUG   CR,"Short Astable Started -- Waiting 5 minutes",CR
CheckAstable2:
                PAUSE   60000                   'wait 5 minutes
                PAUSE   60000                   'wait 5 minutes
                PAUSE   60000                   'wait 5 minutes
                PAUSE   60000                   'wait 5 minutes
                PAUSE   60000                   'wait 5 minutes
AlarmOffEx:
                SEROUT  14,396,[$55,$15]
Done:
                DEBUG   CR,CR,"Done with AN-501"
DoneLoop:
                GOTO    DoneLoop
'Bad Communication Vectors

BadAlarm1:
                DEBUG   "Alarm read 1 bad",CR
                GOTO    ReadAlarmCommand
BadAlarm2:
                DEBUG   "Alarm read 2 bad",CR
                GOTO    ReadAlarmExtended
BadAlarm3:
                DEBUG   "Alarm read 3 bad",CR
                GOTO    ReadAlChar1
BadLoop1:
                DEBUG   "Loop1 error", CR
                GOTO    Loop1
BadReadEx1:
                DEBUG   "Read extended 1 bad",CR
                GOTO    ReadTimeEx1
BadReadEx2:
                DEBUG   "Read extended 2 bad",CR
                GOTO    ReadTimeEx2
```

AN-152 Interfacing with 1 I/O line

The Pocket Watch B uses a two wire serial interface for communication. However, its TM pin is pseudo open-collector. This coupled with the fact that the FM pin and the TM pin never communicate simultaneously allows the Pocket Watch B to communicate via 1 wire if the master processor is able to support this mode.



In order for 1 wire communication to work, the master processor must be able to both send and receive with one I/O line. Because of this, a standard PC serial port is <u>not</u> able to work with just 1 wire. However, most microcontrollers can support this mode. This example shows how to interface the Pocket Watch B to a Parallax BASIC Stamp 1 with only one wire. The diagram to the left shows the connections necessary and the code following gives a brief listing some BSI instructions which will support this method of communication

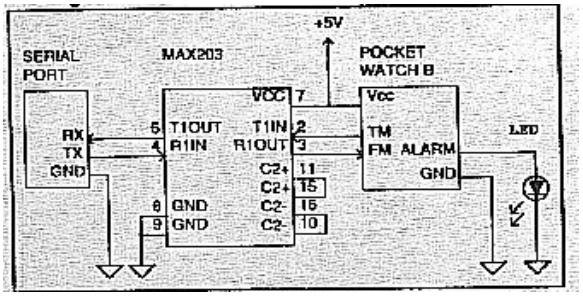
AN-152 Interfacing to a BSI via a single I/O line code listing

```
"AN-152 Single Line BSI interface to Pocket Watch B 'by Solutions Cubed '07/97
'Set I/O pin directions
output 0                'communication TO/FROM Pocket Watch B
input 2                 'ALARM monitoring pin
```

```
Declare variables
Symbol      ss-b0                                                   'seconds
Symbol      mm-b1                                                   'minutes
Symbol      hh-b2                                                   'hours
Symbol      dd-b3                                                   'days
Symbol      mo -b4                                                  'months
Symbol      yl-b5                                                   'years low
Symbol      yh-b6                                                   'years high
Begin:
                    HIGH 0                                          'ensure no spurious start bit
                    PAUSE 1000
SetTimeExtendedCommand:                                             'set to 05:20:00AM, June 7, 1997
                    SEROUT      0,T2400,($55,$10,$00,$14,$05,$07,$06,$61,$13)
SetAlarmExtendedCommand:                                            set to 15 seconds after time
                    SEROUT      0,T2400,($55,$11,$0F,$14,$05,$07,$06,$61,$13)
SetSingleShot:                                                      'Set for 30 seconds
                    SEROUT      0,T2400,($55,$16,$00,$29)
ReadAlarmChar:                                                      'see what time alarm set to go off
                    SEROUT      0,T2400,($55,$19)
                    SERIN       0,T2400,b7,b8,b9,ss,mm,hh,dd,mo,yl,yh
                    DEBUG       CR,"Alarm type: ",#b7,CR
                    DEBUG       "Byle 1: ",#b8,CR
                    DEBUG       "Byle 2: ",#b9,CR
                    DEBUG       "Alarm: ",cr,hh,cr,mm,cr,ss,cr,mo,cr,dd,cr,yh,cr,yl,cr
AlarmOnExtended:
                    SEROUT      0,T2400,($55,$14)
ReadTimeExtended:
                    PAUSE       5000                                'wait 5 seconds between reads
                    SEROUT      0,T2400,($55,$12)
                    SERIN       0,T2400,ss,mm,hh,dd,mo,yl,yh
                    DEBUG       CR,"Time: ",cr,hh,cr,mm,cr,ss,cr,mo,cr,dd,cr,yh,cr,yl,cr
WaitForAlarm:                                                       'wait for single shot to start
                    IF PIN2 = 1 THEN WaitAlarmDone
                    GOTO        ReadTimeExtended
WaitAlarmDone:                                                      'wait for single shot to finish
                    IF PIN2 = 2 THEN Finish
                    GOTO        WaitAlarmDone
Finish:
                    DEBUG       CR,CR,"AN-152 Finished",CR
                    END                                             'ensure no reset
```

AN-153 Interfacing to a PC Serial Port

The Pocket Watch B can be easily interfaced to a PC serial port, thereby allowing any PC program which can control the serial port to interface to the Pocket Watch B. The two wire interface of the Pocket Watch allows for two way communication with no special hardware configuration.



The schematic shows a Maxim 203 RS-232 level translator chip for changing the TTL levels of the Pocket Watch B to RS-232 levels. While there are methods available of interfacing to RS-232 levels without a level translation chip, Solutions Cubed can make no claims as to their reliability.

For a standard PC serial port use the following pin out for the serial connector. While these connections should work with most computers, you should check to make sure.

Signal Name	DB-25	DB-9
TX	2	3
RX	3	2
GND	7	5

Jameco Part number 145630

Miniature Engineering Modules - Pocket Watch B

Distributors

Jameco
1355 Shoreway Road
Belmont, CA 94002-4100
800-831-4242

Marlin P. Jones & Assoc. Inc.
P.O. Box 12685
Lake Park, FL 33403-0685
800-652-6733

Parallax
3805 Atherton Road, #102
Rocklin, CA 95765
888-512-1024

SOLUTIONS CUBED
3029-F Esplanade
Chico, CA 95973
phone (916) 891-8045
fax (916) 891-1643
www.solutions-cubed.com

Disclaimer of Liability and Accuracy: Information provided by Solutions Cubed is believed to be accurate and reliable. However, Solutions Cubed assumes no responsibility for inaccuracies or omissions. Solutions Cubed assumes no responsibility for the use of this information and all use of such information shall be entirely at the user's own risk.
Life Support Policy: Solutions Cubed does not authorize any Solutions Cubed product for use in life support devices and/or systems without express written approval from Solutions Cubed.
Warrantee: Solutions Cubed warrants all Miniature Engineering Modules against defects in materials and workmanship for a period of 90 days. If you discover a defect, we will, at our option, repair or replace your product or refund your purchase price. This warrantee does not cover Miniature Engineering Modules which have been physically abused or misused in any way.

Copyright (c) SOLUTIONS CUBED - 1997. All rights reserved. Version 1.0